WHAT IS CLAIMED IS:

1. A system, comprising:

a client comprising a client Web services stack that supports both a markup language protocol and a binary encoding protocol; and

a server comprising a server Web services stack that supports both the markup language protocol and the binary encoding protocol, wherein the server Web services stack is configured to:

communicate with the client Web services stack according to the markup language protocol; and

dynamically switch to communicate with the client Web services stack according to the binary encoding protocol;

wherein the client Web services stack and the server Web services stack each support the markup language protocol and the binary encoding protocol with a single API.

2. The system as recited in claim 1, wherein the client is a JAX-RPC client.

3. The system as recited in claim 1, wherein the client is a J2ME client.

4. The system as recited in claim 1, wherein the server is a JAX-RPC server.

5. The system as recited in claim 1, wherein the markup language protocol is XML.

6. The system as recited in claim 1, wherein the binary encoding protocol is WS-

Fast.

7. The system as recited in claim 1, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

8. The system as recited in claim 1, further comprising another client comprising another client Web services stack that supports only the binary encoding protocol, and wherein the server Web services stack is further configured to communicate with the other client Web services stack according to the binary encoding protocol.

9. The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server Web services stack is further configured to:

translate the markup language protocol to binary encoding protocol messages for transmission to the client Web services stack; and

translate binary encoding protocol messages received from the client Web services stack to the markup language protocol.

10. The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

11. The system as recited in claim 1, wherein, to communicate with the client Web services stack according to the binary encoding protocol, the server Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.

12. A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a Web services stack configured to:

communicate with another Web services stack on another system according to the markup language protocol; and

dynamically switch to communicate with the other Web services stack according to the binary encoding protocol;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API.

13. The system as recited in claim 12, wherein the system is a JAX-RPC client.

14. The system as recited in claim 12, wherein the system is a J2ME client.

15. The system as recited in claim 12, wherein the system is a JAX-RPC server.

16. The system as recited in claim 12, wherein the system and the other system are peers on a network.

17. The system as recited in claim 12, wherein the markup language protocol is XML.

18. The system as recited in claim 12, wherein the binary encoding protocol is WS-

Fast.

19. The system as recited in claim 12, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

20. The system as recited in claim 12, wherein, to communicate with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to:

translate the markup language protocol to binary encoding protocol messages for transmission to the other Web services stack; and

translate binary encoding protocol messages received from the other Web services stack to the markup language protocol.

21. The system as recited in claim 12, wherein, to communicate with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

22. The system as recited in claim 12, wherein, to communicate with the other Web services stack according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.

23. A system, comprising:

means for communicating between a Web services stack on the system and another Web services stack on another system according to a markup

language protocol; and

means for dynamically switching to communicate between the Web services stack and the other Web services stack according to a binary encoding protocol;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API.

24. The system as recited in claim 23, wherein the markup language protocol is XML, and wherein the binary encoding protocol is WS-Fast.

25. The system as recited in claim 24, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

26. A method, comprising:

a Web services stack communicating with another Web services stack according to a markup language protocol; and

the Web services stack dynamically switching to communicating with the other Web services stack according to a binary encoding protocol;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API.

27. The method as recited in claim 26, wherein the Web services stack is implemented on a JAX-RPC client system.

28. The method as recited in claim 26, wherein the Web services stack is

implemented on a J2ME client system.

29. The method as recited in claim 26, wherein the Web services stack is implemented on a JAX-RPC server system.

30. The method as recited in claim 26, wherein the Web services stack and the other Web services stack are implemented on peers on a network.

31. The method as recited in claim 26, wherein the markup language protocol is XML.

32. The method as recited in claim 26, wherein the binary encoding protocol is WS-Fast.

33. The method as recited in claim 26, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

34. The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises:

translating the markup language protocol to binary encoding protocol messages for transmission to the other Web services stack; and

translating binary encoding protocol messages received from the other Web services stack to the markup language protocol.

35. The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

36. The method as recited in claim 26, wherein said communicating with the other Web services stack according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.

37. A computer-accessible medium comprising program instructions, wherein the program instructions are configured to implement:

a Web services stack communicating with another Web services stack according to a markup language protocol; and

the Web services stack dynamically switching to communicating with the other Web services stack according to a binary encoding protocol;

wherein the Web services stack supports the markup language protocol and the binary encoding protocol with a single API.

38. The computer-accessible medium as recited in claim 37, wherein the Web services stack is implemented on a JAX-RPC client system.

39. The computer-accessible medium as recited in claim 37, wherein the Web services stack is implemented on a J2ME client system.

40. The computer-accessible medium as recited in claim 37, wherein the Web services stack is implemented on a JAX-RPC server system.

41. The computer-accessible medium as recited in claim 37, wherein the Web services stack and the other Web services stack are implemented on peers on a network.

42. The computer-accessible medium as recited in claim 37, wherein the markup language protocol is XML.

43. The computer-accessible medium as recited in claim 37, wherein the binary encoding protocol is WS-Fast.

44. The computer-accessible medium as recited in claim 37, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

45. The computer-accessible medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement:

translating the markup language protocol to binary encoding protocol messages for transmission to the other Web services stack; and

translating binary encoding protocol messages received from the other Web services stack to the markup language protocol.

46. The computer-accessible medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

47. The computer-accessible medium as recited in claim 37, wherein, in said communicating with the other Web services stack according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to

Fast Schema format.

48. A system, comprising:

a processor; and

a memory comprising program instructions, wherein the program instructions are executable by the processor to implement a Web services stack configured to:

communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API;

negotiate with another system to determine if the other system supports the binary encoding protocol;

if the other system supports the binary encoding protocol, communicate with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, communicate with the other system according to the markup language protocol.

49. The system as recited in claim 48, wherein the system is a JAX-RPC server.

50. The system as recited in claim 48, wherein the system and the other system are peers on a network.

51. The system as recited in claim 48, wherein the markup language protocol is XML.

52. The system as recited in claim 48, wherein the binary encoding protocol is WS-Fast.

53. The system as recited in claim 48, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

54. The system as recited in claim 48, wherein the Web services stack is further configured to, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicate with the other system according to the markup language protocol; and

dynamically switch to communicate with the other system according to the binary encoding protocol.

55. The system as recited in claim 48, wherein, to communicate with the other system according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

56. The system as recited in claim 48, wherein, to communicate with the other system according to the binary encoding protocol, the Web services stack is further configured to serialize the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.

57. A system, comprising:

means for communicating with other systems using either a binary encoding protocol or a markup language protocol using a single API;

means for negotiating with another system to determine if the other system supports the binary encoding protocol; and

means for communicating with the other system according to the binary encoding protocol if the other system supports the binary encoding protocol.

58. The system as recited in claim 57, wherein the markup language protocol is XML, and wherein the binary encoding protocol is WS-Fast.

59. The system as recited in claim 57, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

60. A method, comprising:

a Web services stack on a system configured to communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API negotiating with another system to determine if the other system supports the binary encoding protocol;

if the other system supports the binary encoding protocol, the Web services stack communicating with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, the Web services stack communicating with the other system according to the markup language protocol.

61. The method as recited in claim 60, wherein the system is a JAX-RPC server.

62. The method as recited in claim 60, wherein the system and the other system are peers on a network.

63. The method as recited in claim 60, wherein the markup language protocol is XML, and wherein the binary encoding protocol is WS-Fast.

64. The method as recited in claim 60, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

65. The method as recited in claim 60, further comprising, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicating with the other system according to the markup language protocol; and

dynamically switching to communicating with the other system according to the binary encoding protocol.

66. The method as recited in claim 60, wherein said communicating with the other system according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

67. The method as recited in claim 60, wherein said communicating with the other system according to the binary encoding protocol comprises serializing the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.

68. A computer-accessible medium comprising program instructions, wherein the program instructions are configured to implement:

a Web services stack on a system configured to communicate with other systems using either a binary encoding protocol or a markup language protocol using a single API negotiating with another system to determine if the other system supports the binary encoding protocol;

if the other system supports the binary encoding protocol, the Web services stack communicating with the other system according to the binary encoding protocol; and

if the other system does not support the binary encoding protocol, the Web services stack communicating with the other system according to the markup language protocol.

69. The computer-accessible medium as recited in claim 68, wherein the system is a JAX-RPC server.

70. The computer-accessible medium as recited in claim 68, wherein the system and the other system are peers on a network.

71. The computer-accessible medium as recited in claim 68, wherein the markup language protocol is XML, and wherein the binary encoding protocol is WS-Fast.

72. The computer-accessible medium as recited in claim 68, wherein the binary encoding protocol uses Packed Encoding Rules (PER) encoding.

73. The computer-accessible medium as recited in claim 68, wherein the program instructions are further configured to implement, if the other system includes a Web services stack configured to communicate with either the binary encoding protocol or the markup language protocol:

communicating with the other system according to the markup language protocol; and

dynamically switching to communicating with the other system according to the binary encoding protocol.

74. The computer-accessible medium as recited in claim 68, wherein, in said communicating with the other system according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to Fast Infoset format.

75. The computer-accessible medium as recited in claim 68, wherein, in said communicating with the other system according to the binary encoding protocol, the program instructions are further configured to implement serializing the markup language protocol to generate binary encoding protocol messages according to Fast Schema format.